



**Universidad Nacional Mayor de San Marcos**  
**Universidad del Perú. Decana de América**  
Facultad de Ingeniería de Sistemas e Informática  
Escuela Académico Profesional de Ingeniería de Sistemas

## **Grid computing para propósitos científicos**

### **TESINA**

Para optar el Título Profesional de Ingeniero de Sistemas

### **AUTOR**

Richard Nivaldo COPAJA CORNEJO

### **ASESOR**

Norberto Antonio OSORIO BELTRÁN

Lima, Perú

2007

## Referencia bibliográfica

---

Copaja, R. (2007). *Grid computing para propósitos científicos*. Tesina para optar el título de Ingeniero de Sistemas. Escuela Académico Profesional de Ingeniería de Sistemas, Facultad de Ingeniería de Sistemas e Informática, Universidad Nacional Mayor de San Marcos, Lima, Perú.

---

**DEDICATORIA:** A mis padres que con su constante apoyo y confianza me brindan la oportunidad de ser mejor cada día.

## **RESUMEN**

### **GRID COMPUTING PARA PROPOSITOS CIENTIFICOS**

**RICHARD NIVALDO COPAJA CORNEJO**

**SETIEMBRE – 2007**

**Asesor : Lic. NORBERTO OSORIO BELTRÁN**

**Grado : Ingeniero de Sistemas**

---

El presente trabajo tiene como objetivo enfatizar las bondades del Grid Computing en el desarrollo de proyectos de investigación científica.

La evolución de las redes de comunicación de alta velocidad, ha creado un escenario idóneo para el desarrollo de esta tecnología que proporcionará funcionalidades análogas a las existentes en las redes de suministro eléctrico; es decir un único punto de acceso a un conjunto de recursos distribuidos geográficamente como supercomputadores, clusters, sistemas de almacenamiento, fuentes de información, instrumentos y personal.

La tecnología Grid Computing actual ofrece la funcionalidad mínima necesaria para, de forma transparente y segura, compartir y explotar simultáneamente los recursos pertenecientes a diferentes organizaciones,

respetando sus propias políticas y procedimientos de seguridad y gestión de recursos.

La propuesta constituye una solución viable para la difusión y creación de un Grid universitario, a nivel de Lima-Metropolitana, en una primera etapa y en un futuro a nivel nacional.

De esta forma se contribuirá en la elevación del nivel de las investigaciones científicas peruanas.

**Palabras Clave:** Grid, Supercomputación, Arquitectura de Computadores, Sistemas Distribuidos.

## **ABSTRACT**

### **GRID COMPUTING FOR SCIENTIFIC INTENTIONS**

**RICHARD NIVALDO COPAJA CORNEJO**

**SEPTEMBER – 2007**

**Adviser : Lic. NORBERTO OSORIO BELTRÁN**

**Degree : System Engineer**

---

The present work has like objective to emphasize kindness of the Grid Computing in development of scientific research projects.

The evolution of high-speed communications networks has created a appropriate scene for development of this technology that will provide analogous functionalities to the existing ones in electrical networks; that is, an only joining point to collection of resources distributed geographically like supercomputers, clusters, storage systems, information sources, instruments and personnel.

The Grid Computing technology offers necessary functionality to share and operate simultaneously the resources belonging to different organizations, being respected its own policies and security procedures and resource management.

The proposal constitutes a viable solution for the diffusion and creation of university Grid, at Lima-Metropolitan level, in first stage and national level in the future.

It will be a contribution to elevate peruvian scientific researches level.

**Key words:** Grid, Super-computing, Computer Architecture, Distributed System.

## INDICE

RESUMEN	
ABSTRACT	
INDICE .....	1
INDICE DE FIGURAS .....	3
INTRODUCCIÓN .....	4
CAPITULO I .....	6
1. PLANTEAMIENTO DEL PROBLEMA .....	6
1.1 Fundamentación del Problema .....	6
1.1.1 Descripción de la Realidad .....	7
1.1.2 Antecedentes del Problema .....	10
1.1.2.1 Antecedentes en el Perú .....	10
1.1.2.2 Antecedentes en el Mundo .....	11
CAPITULO II .....	14
2. FORMULACIÓN DEL PROBLEMA .....	14
2.1 Objetivos .....	14
2.1.1 Objetivos Generales .....	14
2.1.2 Objetivos Específicos .....	14
2.2 Definición del Problema .....	14
CAPITULO III .....	17
3. MARCO TEORICO CONCEPTUAL .....	17
3.1 Concepto de Grid Computing .....	17
3.1.1 Qué es el Grid .....	18
3.1.2 Lo que el Grid no es .....	20
3.1.3 Beneficios que ofrece el Grid Computing .....	21
3.2 Características del Grid Computing .....	21
3.3 Aplicaciones del Grid Computing .....	22
3.3.1 Supercomputación Distribuida .....	22
3.3.2 Sistemas Distribuidos en Tiempo Real .....	23
3.3.3 Proceso Intensivo de Datos .....	24
3.3.4 Servicios Puntuales .....	25
3.3.5 Entornos Virtuales de Colaboración .....	25
3.4 Arquitectura del Grid .....	25
3.4.1 Capa de Infraestructura .....	26
3.4.2 Capa de Conectividad .....	27
3.4.3 Capa de Recurso .....	28
3.4.4 Capa de Recursos .....	29
3.4.5 Capa de Aplicaciones .....	30
3.5 Intragrid a Intergrid .....	31
3.6 Los Tipos de Grid .....	34



3.6.1	Grid de Información.....	36
3.6.2	Grid de Recursos .....	37
3.6.3	Grid de Servicio.....	39
CAPITULO IV .....		40
4.	METODOLOGÍA DE LA INVESTIGACIÓN .....	40
4.1	Estado del Arte .....	40
4.2	Propuesta de un Grid Universitario .....	41
4.3	Arquitectura a emplear: Open Grid Services Architecture (OGSA)....	41
4.3.2	Open Grid Services Infrastructure (OGSI).....	41
4.3.3	Web Services .....	42
4.3.4	Grid Services.....	43
4.3.4	Arquitectura de OGSA.....	45
4.3.1.1	Capa de los recursos físicos y lógicos.....	45
4.3.1.2	Capa de Web Services .....	45
4.3.1.3	Capa Servicios de Grid de la Arquitectura de OGSA .....	46
4.3.1.4	Capa de Aplicaciones de Grid .....	46
4.4	Herramienta a emplear: Globus Toolkit .....	47
4.4.1	Seguridad.....	47
4.4.2	Componentes de administración de datos .....	48
4.4.3	Administración de recursos .....	48
4.4.4	Servicios de información .....	49
4.5	Arquitectura Conceptual del Grid Universitario.....	50
4.6	Implementación del Grid Universitario .....	52
4.6.1	Integrantes del Grid universitario:.....	52
4.6.2	Estructura del Grid Universitario:.....	52
4.6.2.1	Hardware / Recursos:.....	52
4.6.2.2	Software: .....	52
4.6.3.3	Usuarios / Aplicaciones: .....	53
4.6.3	Etapas del Grid Universitario.....	53
4.6.3.1	Primera Etapa:.....	54
4.6.3.2	Segunda Etapa:.....	55
4.6.3.3	Tercera Etapa:.....	56
4.6.3.4	Cuarta Etapa: .....	56
4.7	Evaluación y Análisis .....	57
CONCLUSIONES .....		58
RECOMENDACIONES .....		59
REFERENCIAS BIBLIOGRÁFICAS.....		60
TERMINOLOGIA.....		62

## **INDICE DE FIGURAS**

Figura 1. Cluster vs. Grid .....	8
Figura 2. Porcentaje del desarrollo de proyectos Grid Computing.....	9
Figura 3. Organizaciones Virtuales.....	20
Figura 4. Supercomputación Distribuida.....	23
Figura 5. Sistemas Distribuidos en Tiempo Real .....	24
Figura 6. Arquitectura de un sistema Grid .....	31
Figura 7. Un Grid más Simple.....	32
Figura 8. Un Intergrid más Complejo.....	34
Figura 9. Los tres tipos de Grid y sus relaciones conjuntas.....	35
Figura 10. Componentes Fundamentales del Globus Toolkit.....	49
Figura 11. Arquitectura Conceptual del Grid Universitario .....	51
Figura 12. Ubicación del Middleware .....	53
Figura 13. Grid UNI – UNMSM .....	54
Figura 14. Grid UNI-UNMSM-UNALM .....	55
Figura 15. Grid Lima-Metropolitana .....	56
Figura 16. Grid Lima-Metropolitana sobre RAAP.....	57

## **INTRODUCCIÓN**

Las actuales tecnologías de Internet se enfocan en la comunicación y el intercambio de información entre computadoras, pero no provee un acercamiento integrado al uso coordinado de recursos en múltiples sitios. El intercambio producido entre organizaciones está enfocado a compartir información frecuentemente a través de servidores centralizados.

Dado este escenario, el Grid Computing ha emergido como un importante nuevo campo, diferenciado de la computación distribuida convencional al establecer su foco principalmente en compartir recursos. El Grid Computing permite, que una organización aproveche el poder de procesamiento y almacenamiento de múltiples computadoras, posiblemente distribuidas a nivel mundial, trabajando de forma coordinada y así lograr un poder similar al de un supercomputador.

A través de una red como la Internet, se distribuye el trabajo a múltiples máquinas y se aprovecha los recursos subutilizados para realizar una tarea. De esta forma se cuenta con infraestructuras mundiales que son compartidas por científicos dispersos en todo el planeta.

Este modelo de cómputo se utiliza en entornos de investigación desde la década de los 90 en el ámbito mundial.

En el Perú, no existe mucho conocimiento acerca del Grid Computing es por eso que el siguiente trabajo presenta una propuesta de implementar un Grid universitario, para dar a conocer y promover este paradigma de la computación distribuida aplicándolo a propósitos científicos y académicos.

El capítulo 1 contiene el planteamiento del problema a través de la fundamentación del problema, mediante una descripción de la realidad y sus antecedentes tanto a nivel nacional e internacional.

El capítulo 2 contiene la formulación del problema basado en sus objetivos para así definir el problema.

El capítulo 3 contiene las generalidades del marco teórico necesarios para plantear las bases teóricas.

El capítulo 4 contiene la propuesta metodológica, la cual contempla el desarrollo de un Grid universitario en varias fases.

Al final se muestra las conclusiones, recomendaciones, referencias bibliográficas y terminología para el entendimiento adecuado del presente trabajo.

## **CAPITULO I**

### **1. PLANTEAMIENTO DEL PROBLEMA**

#### **1.1 Fundamentación del Problema**

El problema del gran consumo de recursos de muchos proyectos científicos de gran envergadura, impone una restricción para las organizaciones que los impulsan en la realización de sus objetivos.

Un claro ejemplo, es la capacidad de procesamiento requerida en ambiciosos proyectos de investigación científica, simulaciones a gran escala, toma de decisiones a partir de grandes volúmenes de información y cientos de casos imaginables que no encuentran una solución, o quizá parte de ella, en las herramientas disponibles en la tecnología actual.

En la actualidad, existe una tecnología emergente conocida como Grid Computing la cual permite que una organización aproveche el poder de procesamiento y almacenamiento de múltiples computadoras, trabajando de forma coordinada y así poder lograr un poder similar al de un supercomputador.

El beneficio del uso de esta tecnología es un proceso más rápido y eficaz en trabajos computo-intensivos, utilizando los recursos existentes y eliminando la desventaja de ligar máquinas específicas herméticamente a trabajos específicos, permitiendo un pool de agregados para realizar trabajos secuenciales o paralelos.

### **1.1.1 Descripción de la Realidad**

La utilización de los recursos en los proyectos científicos y como estos se gestionan desde los Clusters, tienen una gran diferencia si se gestionarían desde Grid Computing, estas diferencias se muestran en la figura 1.

**Clusters:** el reparto de recursos se realiza de modo centralizado, y todos los nodos trabajan cooperativamente como un único recurso unificado.

**GRID:** cada nodo tiene su propio gestor de recursos, y no pretender dar una única visión del sistema.

	<b>Cluster</b>	<b>Grid</b>
<b>Nodos</b>	Grandes computadores	Computadoras de escritorio
<b>Propiedad</b>	Única	Múltiple
<b>Usuarios</b>	Centralizados	Descentralizados
<b>Recursos</b>	Centralizados	Distribuidos
<b>Planificación</b>	Centralizada	Descentralizados
<b>Imagen única</b>	Sí	No
<b>Escalabilidad</b>	100s	>1000s
<b>Capacidad</b>	Garantizada	Variable, alta
<b>Rendimiento</b>	medio	alto

Figura 1. Cluster vs. Grid

Fuente: [http://www.jatit.org/research/introduction\\_grid\\_computing.htm](http://www.jatit.org/research/introduction_grid_computing.htm)

Son pocas las instituciones científicas o académicas en el Perú que desarrollen o formen parte de proyectos de Grid Computing, a excepción del Servicio Nacional de Meteorología e Hidrología del Perú (SENAMHI) que fue partícipe del proyecto EELA, proyecto que busca difundir el uso de tecnologías de Grid Computing entre organizaciones europeas y latinoamericanas.

A nivel mundial el mayor porcentaje de proyectos, de Grid Computing, son desarrollados en países donde se aprecia un mayor avance y uso de la computación distribuida. Como se muestra en la figura 2, EEUU es el país donde hay más desarrollo de proyectos utilizando Grid Computing, siguiéndole muy de cerca los países europeos, los cuales de alguna manera fomentan e incentivan el uso del Grid Computing en países Latinoamericanos.

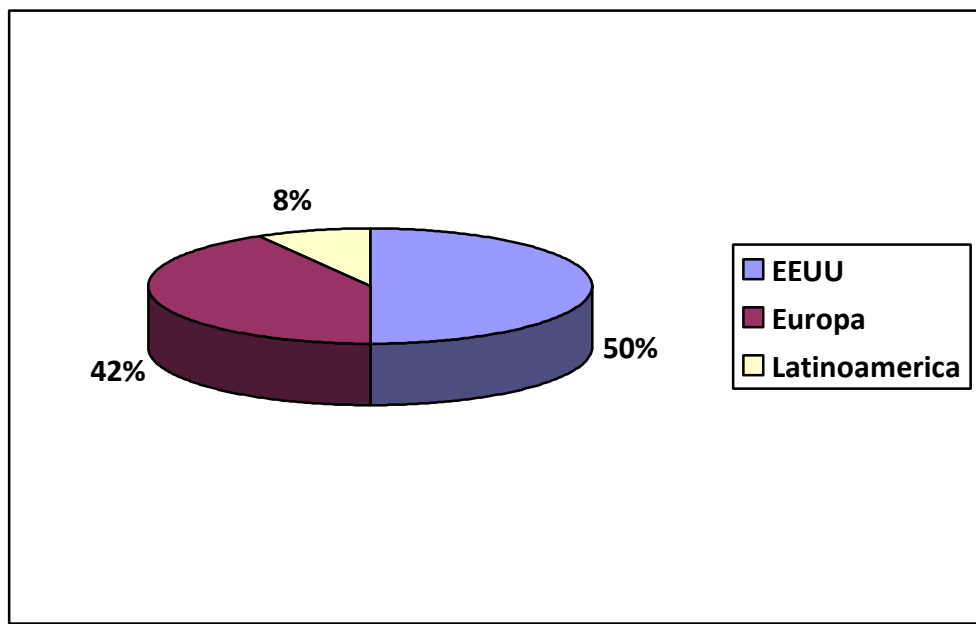


Figura 2. Porcentaje del desarrollo de proyectos Grid Computing

Fuente: <http://www.rediris.es/rediris/boletin>



## **1.1.2 Antecedentes del Problema**

### **1.1.2.1 Antecedentes en el Perú**

#### **Caso SENAMHI, EELA y la predicción climática**

El cambio climático puede producir enormes trastornos en el sistema natural del Perú, ante esta situación SENAMHI se había planteado la posibilidad de desarrollar un sistema de computación Grid local, a fin de poder obtener una mayor rapidez al ejecutar su aplicación de predicción climática.

En el transcurso del proyecto tomaron conocimiento de una convocatoria europea para desarrollar proyectos de investigación y tomaron contacto con la Universidad de Cantabria para participar del proyecto de predicción climática.

De esta manera SENAMHI se integro al proyecto EELA e inicio el trabajo para portar su aplicación de predicción climática para que fuera compatible con la arquitectura Grid.

Ejecutar la aplicación utilizando los datos climáticos de una semana con un cluster de 25 Gigafllops demoraría entre cuatro a cinco horas aproximadamente; sin embargo, al ejecutar la misma aplicación conectada al Grid se obtiene resultados en tan sólo 90 minutos.

### **1.1.2.2 Antecedentes en el Mundo**

#### **Caso SETI@home y la búsqueda de vida extraterrestre.**

SETI@home es un proyecto científico que consiste en el procesamiento de señales de radio para buscar una prueba de inteligencia extraterrestre; fue el primer intento de computación distribuida realizado con éxito y en el que participan voluntarios de todo el mundo.

Una vez instalado en la PC analiza señales procedentes del espacio en busca de señales de inteligencia extraterrestre; estas señales son captadas por el radiotelescopio de Arecibo, en Puerto Rico, que con sus 305 metros de diámetro es el mayor del mundo, lo que permite recoger señales mucho más débiles que cualquier otro radiotelescopio.

Estas señales se envían a la Universidad de Berkeley, California, donde se dividen en fragmentos muy pequeños que son repartidos entre los más de 4 millones de usuarios inscritos en el proyecto SETI@home

Para analizar toda la información recogida en Arecibo en un sólo computador, se necesitaría que este fuera el más grande jamás construido, y su coste lo haría impracticable para un proyecto de estas características que no busca un interés económico, así que los fundadores del proyecto pensaron que habiendo millones de PCs conectados a Internet alrededor de todo el mundo, estos podrían colaborar de alguna manera.

### **Caso Great Internet Mersenne Prime Search**

Es un proyecto mundial de investigación llamado Great Internet Mersenne Prime Search (GIMPS), que ha descubierto el mayor número primo conocido utilizando su PC y software producido por George Woltman y Entropia, Inc. como parte de la Grid internacional de más de 205,000 computadoras en conexión impulsada por la firma. El nuevo número, caracterizado como  $2^{13,466,917}-1$ , consta de 4,053,946 dígitos. Pertenece a una clase especial de números primos llamada primos de Mersenne. El descubrimiento representa sólo el número 39 de los primos de Mersenne conocidos, estos números son llamados así por Marin Mersenne, un monje francés del siglo XVII que fue el primero en estudiarlos. Un primo de Mersenne es un primo de la forma  $2^p-1$ . Los primeros primos de Mersenne son 3, 7, 31, 127, etc. Estos primos son muy relevantes para la teoría de los números, encriptación y desencriptación de mensajes y también han sido utilizados para pruebas de hardware entre otros.

### **Caso FightAIDSatHome y la lucha contra el SIDA**

El proyecto FightAIDSatHome, desarrollado por una alianza entre Entropia y Olson Laboratory, consiste en desarrollar medicamentos para combatir el SIDA. Para desarrollar estos medicamentos es necesario modelar los virus y los componentes que formarán estos medicamentos para desarrollar curas más efectivas. Todo esto es posible por medio de supercomputadoras astronómicamente costosas, pero mediante este proyecto de indudable ingenio es posible donar tiempo y recursos ociosos de la computadora de un usuario la cual formará parte de una extensa red de donantes que realizara millones de

cálculos con los recursos ociosos de estas computadoras y de esta manera acelerar el proceso de obtención de estos medicamentos.

En el programa están inscritas 42.700 computadoras que han generado hasta el momento casi 21 millones de horas de trabajo y con una capacidad disponible de 8,8 Terabytes de memoria y un espacio en disco de casi 1.000 Teras.

## **CAPITULO II**

### **2. FORMULACIÓN DEL PROBLEMA**

#### **2.1 Objetivos**

##### **2.1.1 Objetivos Generales**

Describir y Evaluar las ventajas del uso del Grid Computing en proyectos de investigación científica que presentan el problema del alto consumo de recursos.

##### **2.1.2 Objetivos Específicos**

Localizar los proyectos que necesiten mayor infraestructura para poder continuar con sus investigaciones, estos proyectos podrían ser a nivel nacional.

Formar unidades de investigación que permitan definir las necesidades de cada proyecto de investigación científica.

#### **2.2 Definición del Problema**

La necesidad de recursos en los proyectos científicos hace que estos no logren sus objetivos o los realizan en periodos mayores a los esperados, ante esta necesidad surgen nuevos modelos de computación distribuida.

Hasta hace poco, restringida a los ámbitos académicos y de investigación, como en su momento ocurrió con Internet y la World Wide Web (WWW), Grid empieza a salir del cascarón perfilándose como una prometedora alternativa informática distribuida de área amplia en el campo científico y de investigación.

Desde mediados de los 90, en que el término Grid fue acuñado para designar esta propuesta de informática distribuida para la investigación científica y la ingeniería avanzada que pudiera aprovechar la omnipresencia de Internet, los países con mayores avances en TI, como es el caso de EEUU y los países europeos, vienen aplicando en sus proyectos de investigación científica y académica este nuevo paradigma, logrando de esta manera una mayor eficiencia en menores tiempos y sin incrementar su inversión en recursos.

En muchos países Latinoamericanos, como es el caso de Perú, existe poco avance, conocimiento y uso del Grid Computing, solo algunos como Chile, Brasil, Argentina y Venezuela vienen trabajando en diversos proyectos utilizando Grid.

En el Perú, no existen muchos proyectos de investigación por la necesidad de una alta inversión en infraestructura y la falta de conocimiento de tecnologías como el Grid Computing. Son pocas las entidades que se han unido para desarrollar este tipo de tecnología, en comparación con nuestros pares latinoamericanos que ya vienen trabajando con computación distribuida, siendo SENAMHI la pionera en nuestro país.

Es por ello, que el presente trabajo pretende fomentar y describir las ventajas del uso y desarrollo del Grid Computing en proyectos de investigación científica, que presentan el problema del alto consumo de recursos en la realización de procesos para que puedan cumplir sus objetivos planteados y continuar con sus investigaciones.

Para ubicar proyectos en el ámbito nacional que necesiten mayor infraestructura se propone formar unidades de investigación que permitan precisar las necesidades de cada una de ellas y definan cómo el Grid puede cubrir dichas necesidades para hacer que el proyecto sea viable y cumpla con sus objetivos de manera eficaz y eficiente.

## **CAPITULO III**

### **3. MARCO TEORICO CONCEPTUAL**

#### **3.1 Concepto de Grid Computing**

La utilización en paralelo de la capacidad computacional de los diferentes sistemas que conforman un Grid Computing, permite realizar las tareas cotidianas de trabajo mucho más rápido que si de una única computadora se tratara.

Mientras que la Web es un servicio para compartir información a través de Internet, el Grid Computing es un servicio para compartir potencia de cálculo y capacidad de almacenamiento a través de la red. Esta compartición se realiza virtualizando los recursos que participan en una infraestructura de Grid Computing, de manera que para el usuario final actúan como un único y potente computador.

Los teóricos del Grid Computing entienden que el objetivo final de la tecnología Grid Computing es crear una infraestructura cuyo ámbito sea todo Internet, integrando todos los recursos heterogéneos computacionales que existen alrededor del mundo [4].



El concepto principal de Grid Computing es el de compartir potencia computacional [2].

Desde el momento en el que las primeras computadoras comenzaron a conectarse a Internet, surgió la idea de unir la potencia inutilizada de cada uno para abordar problemas a los que sólo podían enfrentarse las supercomputadoras pertenecientes a organizaciones gubernamentales, universidades o grandes multinacionales.

La tecnología que hace esto posible se llama Grid Computing y se basa en el aprovechamiento de los ciclos de procesamiento no utilizados por los millones de computadores conectados a la red, logrando de esta forma la realización de tareas que son demasiado intensivas para ser resueltas por una única computadora.

### **3.1.1 Qué es el Grid**

El Grid permite la Computación Distribuida sobre de una red de recursos heterogéneos utilizando estándares abiertos [5].

Un sistema Grid puede entenderse, en una primera aproximación, como un conjunto de equipos informáticos geográficamente distribuidos que colaboran de forma conjunta para obtener un fin común. Estos recursos pueden pertenecer a empresas diferentes y su ubicación geográfica puede variar substancialmente, pero pueden considerarse sistemas del mismo Grid equipos separados unos pocos metros u otros localizados en extremos opuestos al planeta.

Su definición varía poco respecto a los tradicionales y ampliamente conocidos Sistemas Distribuidos, no obstante esto no nos debe llevar a un engaño. Uno de los valores diferenciales de los sistemas Grid es el término recurso, que va más allá de la simple compartición o intercambio de archivos entre equipos. Abarca el acceso directo a computadores CPU, software, datos o cualquier otro elemento disponible y siempre dentro de una estrategia colaboradora con el fin de solucionar un problema concreto. El uso compartido de recursos pertenecientes a diferentes compañías da lugar a lo que se conoce como Virtual Organizations (Organizaciones Virtuales, VO) [2].

En esencia, el Grid es una estructura de recursos que se virtualizan para optimizar los propósitos. Esto significa mayor poder de CPU, aplicaciones, datos, almacenamiento y recursos de red. Está para que el usuario la utilice cuando sea necesario, sin preocuparse de dónde se localiza, en qué sistema operativo residen los datos y demás recursos, etc.

Una organización real puede participar en una o más VOs compartiendo algunos o todos sus recursos. En la Figura 3, se muestran tres organizaciones reales y dos VOs. Las políticas que controlan el acceso a los recursos varían de acuerdo a las organizaciones reales, los recursos y las VOs involucradas.

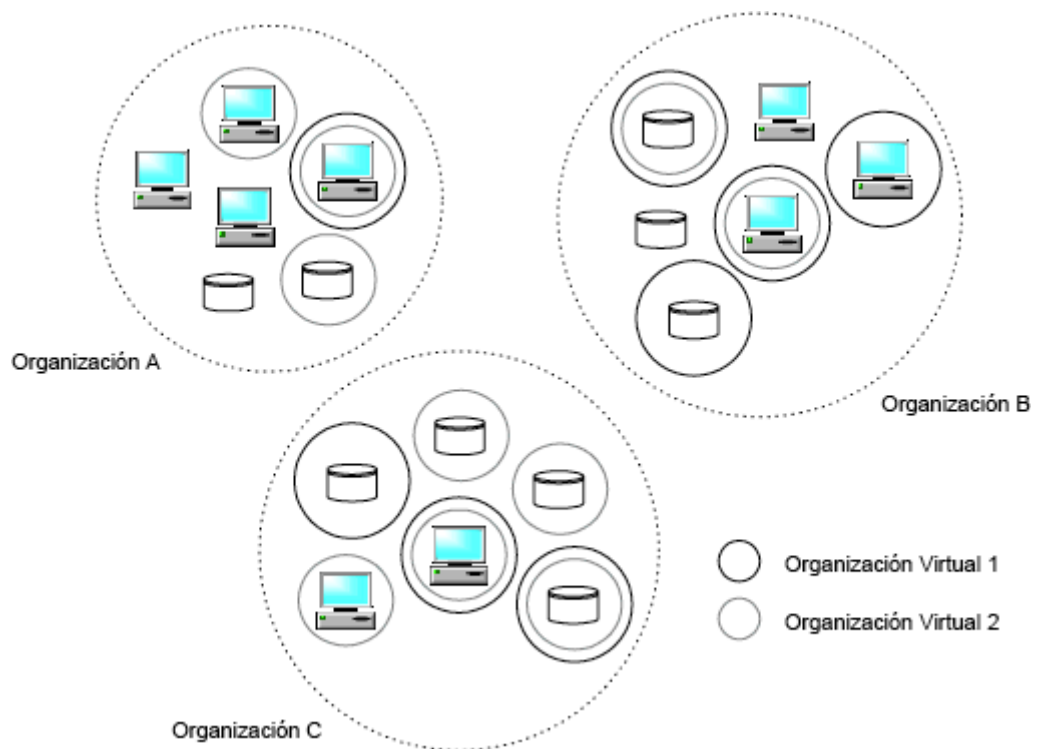


Figura 3. Organizaciones Virtuales.

Fuente: <http://www.textoscientificos.com/redes/computacion-grid>

### 3.1.2 Lo que el Grid no es

- NO es una mejora/ampliación de Internet (no están al mismo nivel).
- NO es un proyecto (es una tecnología).
- NO es un cluster de computadoras (en un Grid puede haber integrados muchos o ningún cluster) [4].

### **3.1.3 Beneficios que ofrece el Grid Computing**

Los principales beneficios son:

- Ofrecer flexibilidad para llenar las necesidades cambiantes del negocio.
- Brindar alta calidad a menor costo.
- No necesitar de toda una nueva infraestructura para que funcione.
- Facilitar poder de computación / precio muy barato.
- Brindar el poder de un supercomputador.
- Utilizar software gratuito y usar código fuente abierto.
- No precisar hardware adicional, para posibilitar el incremento de la potencia de cómputo.
- Brindar transparencia para el usuario que participa en el Grid.
- Disponer de mecanismos de seguridad y autenticación.[7]

## **3.2 Características del Grid Computing**

Un Grid Computing es una infraestructura hardware y software que suministra al que la utiliza acceso seguro (dependable), consistente (consistent), penetrante (pervasive) y barato (inexpensive), a elevadas capacidades computacionales [2].

El concepto de infraestructura se utiliza porque un Grid es un conjunto de recursos (ciclos de CPU, datos, sensores, etc.), y todos esos recursos necesitan una interconexión de hardware y un control de software para que estén ensambladas en un Grid.

Esta infraestructura debe proporcionar a los usuarios un servicio seguro a todos los niveles: capacidad de cómputo, de integridad de datos, de seguridad de acceso, etc.

El servicio debe ser consistente, basado en estándares, y de esta manera el acceso y las operaciones sobre el Grid estarán definidas por dichos estándares evitando la heterogeneidad.

La idea de penetración no es tanto la posibilidad de acceder a cualquier recurso del Grid como que el Grid llega a cualquier sitio, de esta manera se asegura que una vez conectado desde cualquier punto se puede extraer del Grid toda la potencia que se requiera.

Por último el acceso y uso del Grid debe tener un coste económico que le haga atractivo para que su utilización se universalice.

### **3.3 Aplicaciones del Grid Computing**

El análisis lleva a definir cinco grandes áreas de trabajo determinadas por las necesidades de cálculo, espacio para el almacenamiento de los datos y tiempo de respuesta [7]. Las áreas son:

#### **3.3.1 Supercomputación Distribuida**

Dentro de esta área, se encuentran aquellas aplicaciones cuyas necesidades son imposibles de satisfacer en un único nodo. Estas necesidades se producen en instantes de tiempo determinados y consumen muchos recursos, por lo que se dice que son puntuales e intensivas.

Ejemplo de este tipo de aplicaciones son las simulaciones, las herramientas de cálculo numérico, los procesos de análisis de datos, la extracción de conocimiento de almacenes de datos, etc. Ver figura 4.

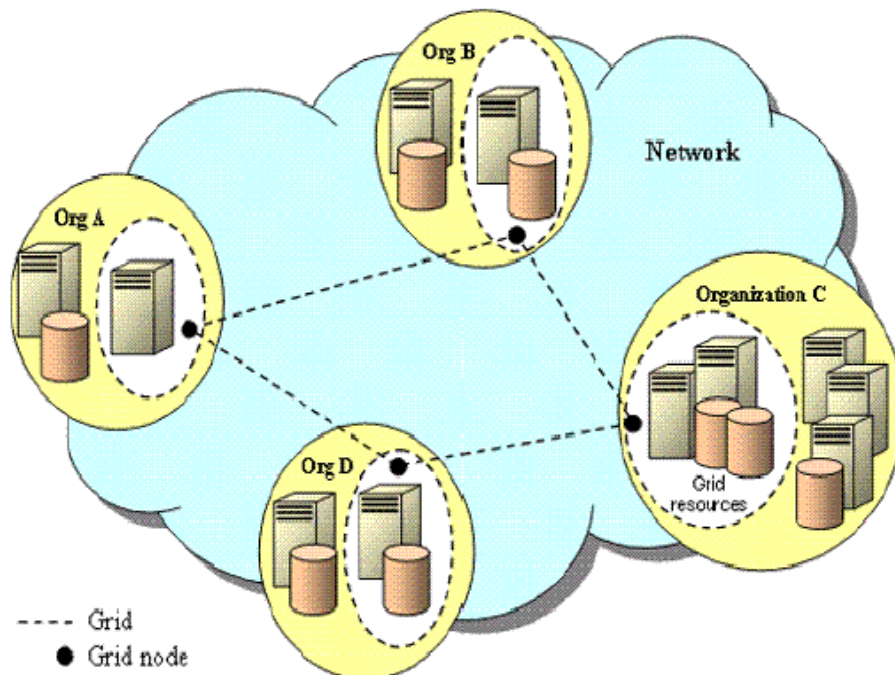


Figura 4. Supercomputación Distribuida.  
Fuente: [www.uae.ma/uaegrid/gridcomputing](http://www.uae.ma/uaegrid/gridcomputing)

### 3.3.2 Sistemas Distribuidos en Tiempo Real

En este tipo de aplicaciones se consideran aquellas que generan un flujo de datos a alta velocidad que debe ser analizado y procesado en tiempo real. Ejemplo de este tipo de aplicaciones son los experimentos de física de alta energía, control remoto de equipos médicos de alta precisión y precio, todos los procesos de la denominada e-Medicine, el tratamiento de imágenes para la visión artificial, etc. Ver figura 5.

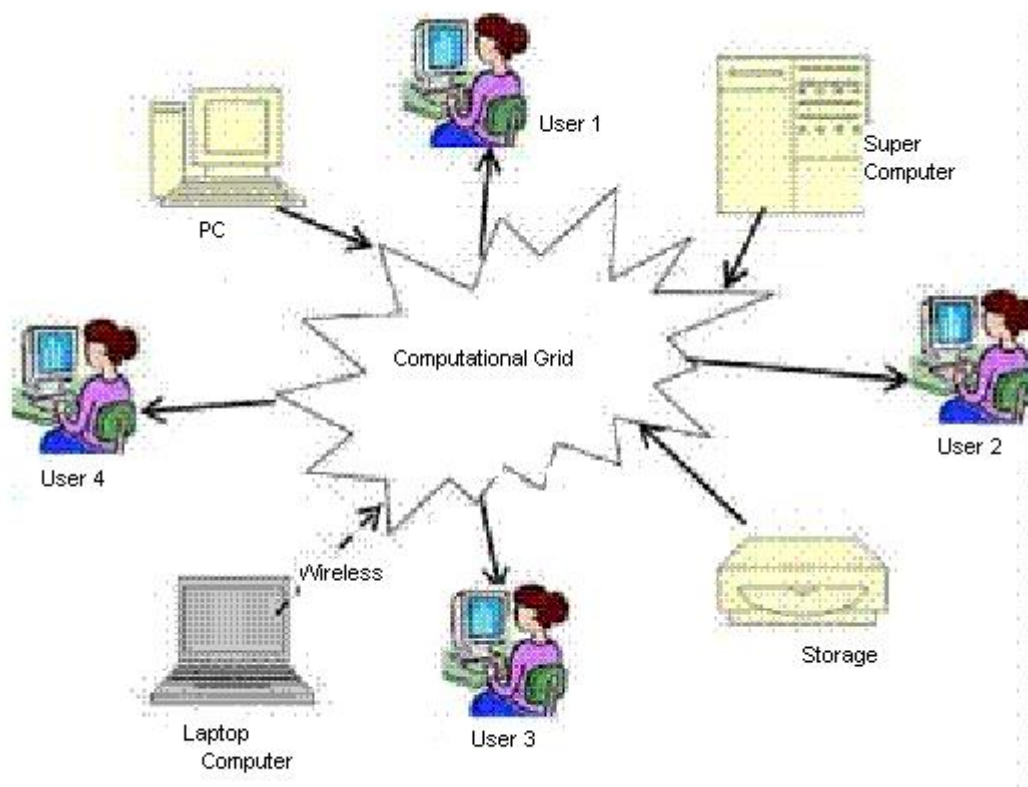


Figura 5. Sistemas Distribuidos en Tiempo Real

Fuente: [www.textoscientificos.com/redes/computacion-grid](http://www.textoscientificos.com/redes/computacion-grid)

### 3.3.3 Proceso Intensivo de Datos

Esta área se centra en aquellas aplicaciones que hacen un uso intensivo del espacio de almacenamiento. Las necesidades de almacenamiento de este tipo de aplicaciones desbordan la capacidad de almacenamiento de un único nodo y los datos son distribuidos por todo el Grid. Además de los beneficios por el incremento de espacio, la distribución de los datos a lo largo del Grid permite el acceso a los mismos de forma distribuida. Ejemplos de este tipo de aplicaciones son todos los sistemas gestores de bases de datos distribuidas.

### **3.3.4 Servicios Puntuales**

En esta área, se olvida el concepto de potencia de cálculo y capacidad de almacenamiento, para centrarse en recursos que una organización puede considerar como no necesarios. De esta manera el Grid ofrece a la organización esos recursos sin que la organización deba desarrollarlos por si misma. Ejemplos de este tipo de aplicaciones son aquellas que permiten acceder a hardware muy específico (equipos costosos de medida o de análisis de muestras) para la realización de labores a distancia.

### **3.3.5 Entornos Virtuales de Colaboración**

Esta área está relacionada directamente con el concepto de Tele inmersión, de manera que se utilizan los enormes recursos computacionales del Grid y su naturaleza distribuida para generar entornos virtuales 3D distribuidos.

## **3.4 Arquitectura del Grid**

Podemos describir la arquitectura Grid en términos de capas [2] donde cada una proveerá funciones específicas. Las capas más altas de esta estructura se enfocan y acercan más a la relación directa con el usuario, mientras que las del nivel más bajo se aproximan al hardware, es decir a las computadoras y redes, estas capas se muestran en la Figura 6.



### 3.4.1 Capa de Infraestructura

Provee los recursos en los que el acceso compartido es mediado por protocolos. Un recurso puede ser una entidad lógica, un sistema de archivo distribuido, un cluster de computadoras o un pool de computadoras distribuido, en esos casos la implementación del recurso puede involucrar protocolos internos.

La funcionalidad de esta capa permite el sharing de operaciones sofisticadas.

Los siguientes detalles de los recursos específicos presentes en esta capa:

- **Recursos computacionales:** mecanismos requeridos para iniciar programas, controlar y monitorear la ejecución de procesos resultantes. El manejo de mecanismos que permite el control sobre los recursos alojados a procesos son usados como mecanismos avanzados de reserva. Se necesitan algunas funciones de evaluación para la determinación de características de hardware y software así como información de estado relevante.

- **Recursos alojados:** se requieren algunos mecanismos para el guardado y recuperación de archivos. Hay mecanismos para la lectura y escritura de subconjuntos de archivos y/o ejecución remota de una selección de datos o reducción de funciones. El manejo de estos mecanismos permite el control sobre los recursos alojados, para transferencia de datos y otras funciones específicas.

- **Recursos de red:** mecanismos de manejo que provean control sobre los recursos y la transferencia en la red, funciones que provean las características de red y carga.

- **Repositorios de recursos:** es una forma especializada en almacenamiento de recursos que requiere mecanismos de manejo de versiones de fuentes y código de objetos.

- **Catálogos:** forma especializada de almacenamiento que requiere mecanismos para implementar consultas y actualización de operaciones (ej. BD relacionales).

### 3.4.2 Capa de Conectividad

Esta capa define el centro de las comunicaciones y autenticación de protocolos requeridos para transacciones específicas de Grid dentro de la red. Los protocolos de comunicación proveen el intercambio de datos entre los recursos de la capa de infraestructura. Los protocolos de autenticación construyen la comunicación de servicios para proveer mecanismos seguros para la verificación e identificación de usuarios y recursos.

Los requerimientos de comunicación requieren transporte, ruteo, y servicio de nombre (TCP, DNS, IP e ICMP). Las soluciones de autenticación para VOs deben tener las siguientes características:

- **Single sign on:** Los usuarios debe ser capaces de autenticarse solo una vez y luego tener acceso a múltiples recursos Grid definidos en la capa de infraestructura.

- **Delegación:** Un usuario debe ser capaz de proveer un programa con la habilidad de acceder a recursos sobre los cuales el usuario está autorizado.

- **Integración con varias soluciones de seguridad local:** Las soluciones Grid deben ser capaces de interoperar con varias soluciones locales empleadas por cada sitio o proveedor de recurso.

- **Relaciones de usuarios basadas en la confianza:** Si un usuario utiliza recursos de múltiples proveedores a la vez, el sistema de seguridad no debe requerir que cada proveedor de recursos coopere o interactúe con cada una de las configuraciones de los entornos de seguridad.

### **3.4.3 Capa de Recurso**

La capa de Recurso construye sobre la capa de Conectividad, protocolos de Comunicación y Autenticación para definir negociaciones seguras, iniciación, monitoreo, control, contabilidad y pagos de operaciones compartidas sobre recursos individuales. Las implementaciones de estos protocolos llaman a funciones de la capa de infraestructura para acceder y controlar recursos locales. Los protocolos de esta capa se refieren a recursos individuales y por ende ignoran resultados de estado global y acciones atómicas a lo largo de colecciones distribuidas, éstos son de consideración para la capa colección. Existen dos tipos de protocolos en esta capa:

- **Protocolo de información:** para obtener información sobre la estructura y estado de los recursos.

- **Protocolos de administración:** son utilizados para negociar acceso a recursos compartidos, especificando, por ejemplo requerimientos de recurso como calidad de servicio y las operaciones a ser ejecutadas. También debe soportar la monitorización del estado de una operación y control de la misma. Debe además asegurar que las operaciones de protocolos peticionadas sean consistentes con la política bajo la cual el recurso debe ser compartido.

#### **3.4.4 Capa de Recursos**

Esta capa contiene protocolos y servicios que no están asociados con ningún recurso específico, sino son globales en naturaleza y capturan interacciones a través de colecciones de recursos. Permiten implementar una amplia variedad de comportamientos al compartir, sin establecer nuevos requerimientos sobre los recursos que están siendo compartidos. Podemos mencionar algunos ejemplos:

- **Servicio de directorio:** permiten a los participantes de una VO descubrir la existencia y/o propiedad de los recursos de una VO, y puede consultarse por tipo de recurso, atributo, disponibilidad o carga.

- **Ubicación, planificación:** permite a los participantes de la Organización Virtual requerir la ubicación de uno o más recursos para un específico propósito, y la planificación de tareas sobre los recursos apropiados.

- **Servicios de diagnóstico y monitoreo:** soportan el monitoreo de recursos de la VO por fallas, ataques, sobrecargas, y demás.

- **Servicios de replicación de datos:** soportan la administración de recursos de almacenamiento de las VOs, para maximizar el rendimiento al acceso de datos con respecto a métricas tales como tiempo de respuesta, confianza, y costo.

- **Servicios de colaboración:** Soportan un coordinado intercambio de información dentro de comunidades de usuarios potencialmente grandes, sea sincrónica o asincrónicamente.

### **3.4.5 Capa de Aplicaciones**

La última capa en la arquitectura Grid comprende las aplicaciones de usuario que operan dentro del ambiente de la VO. Las aplicaciones son construidas en términos de servicios definidos en cualquier capa. En cada capa, hay protocolos bien definidos que proveen acceso a algún servicio útil: administración de recursos, descubrimiento de recursos, acceso a datos, y demás.

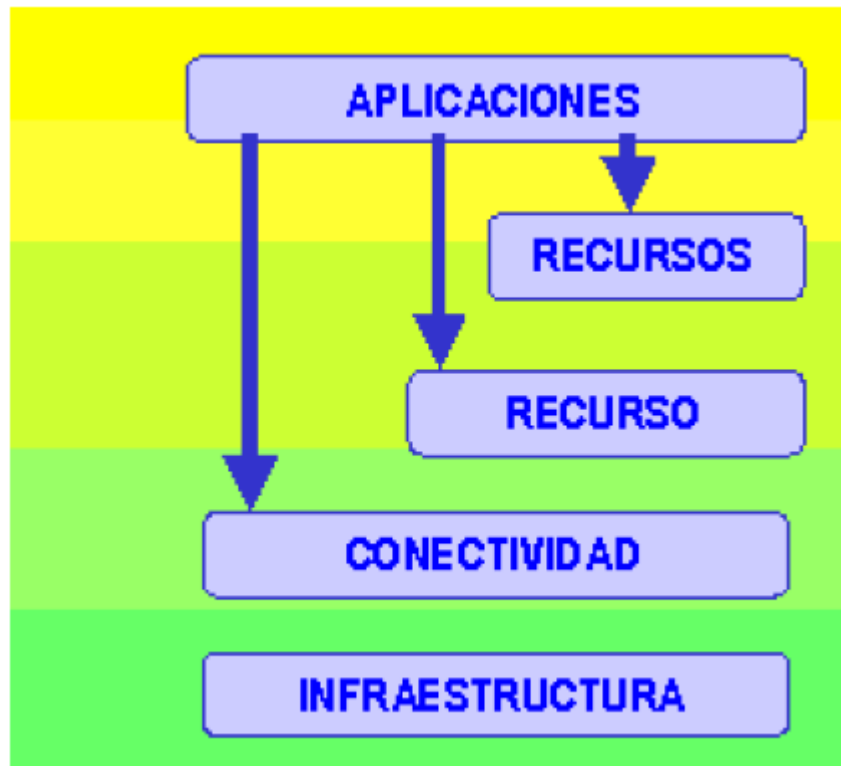


Figura 6. Arquitectura de un sistema Grid

Fuente: <http://exa.unne.edu.ar/depar/areas/informatica/SistemasOperativos/Grid.pdf>

### 3.5 Intragrid a Intergrid

Como se presenta en figura 7, el Grid más simple consiste en sólo unas pocas máquinas, todas de la misma arquitectura de hardware y el mismo sistema operativo, conectadas en una red local. Este tipo de Grid usa sistemas homogéneos, así hay menos consideraciones y pueden ser usados sólo para experimentar con el software de Grid [1].

Las máquinas usualmente están en un departamento de una organización, y sus usos como un Grid pueden no requerir una política especial de seguridad. Debido a que las máquinas tienen la misma arquitectura y sistema operativo, elegir software de aplicación para dichas máquinas es

generalmente sencillo. Esto podría ser llamarlo una aplicación de cluster en lugar de Grid.

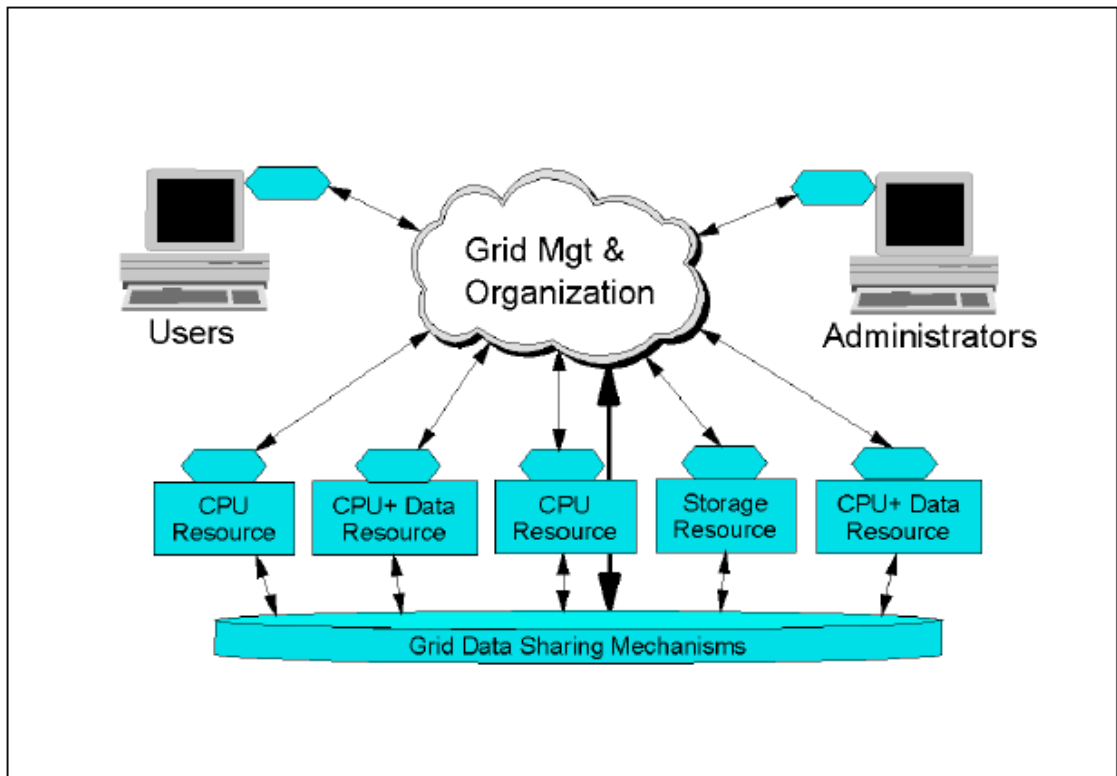


Figura 7. Un Grid más Simple.

Fuente: Fundamentals of Grid Computing.

El siguiente avance sería incluir las máquinas heterogéneas. En esta configuración, hay más tipos de recursos disponibles. El sistema Grid puede incluir algunos componentes de scheduling. También puede lograrse el compartimiento de archivos, usando los sistemas de archivo de red.

Las máquinas que participan en el Grid pueden incluir uno de los departamentos de la misma organización. Tal modelo de Grid también será llamado “Intragrid”.

Cuando el Grid se extiende a muchos departamentos, las políticas pueden requerirse para indicar cómo el Grid debe usarse. Por ejemplo puede haber políticas, para qué tipos de trabajo permite el Grid y cuántas veces.

Puede haber también una priorización por departamento o por tipos de aplicaciones que deben tener acceso a los recursos del Grid.

Los datos sensibles en un sólo departamento pueden necesitar ser protegido contra el acceso de trabajos que se ejecutan para otros departamentos.

Las máquinas del Grid especializadas pueden ser agregadas para aumentar la calidad de servicio, en lugar de depender completamente de los recursos de barrido.

El Grid puede crecer geográficamente en una organización que tiene los medios en diferentes ciudades. Las conexiones de comunicaciones dedicadas pueden usarse entre estas facilidades y el Grid.

En algunos casos, VPN tunneling u otras tecnologías pueden usar Internet para conectar las diferentes componentes de la organización. La seguridad aumenta una vez que los límites de cualquier facilidad se superan.

El Grid puede crecer para ser jerárquicamente organizado para reducir la contención implicada por el control central, aumentando la escalabilidad.



Un Grid puede crecer, cruzar los límites de la organización y puede usarse para colaborar en los proyectos de interés común. Esto se conoce como un “Intergrid” (como se ve en la figura 8).

Habitualmente se requieren los niveles más altos de seguridad en esta configuración para prevenir posibles ataques y espionajes [1].

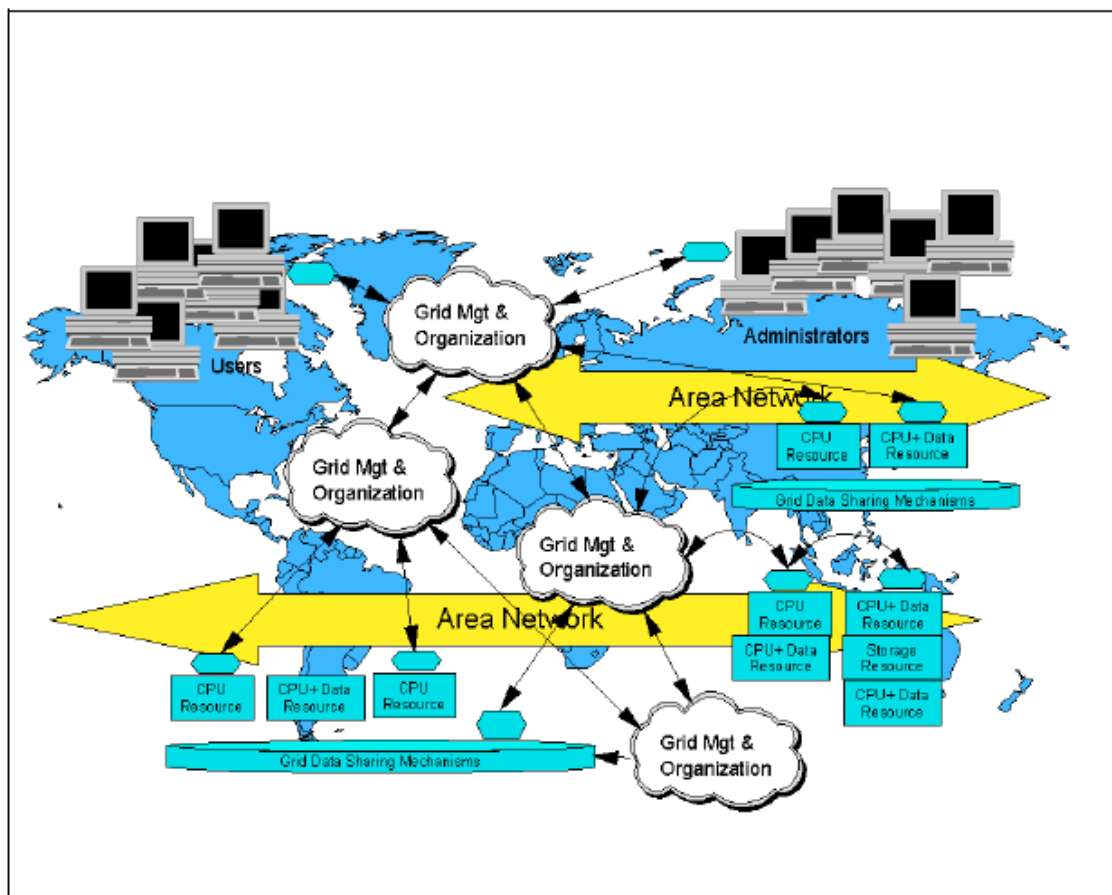


Figura 8. Un Intergrid más Complejo.  
Fuente: Fundamentals of Grid Computing.

### 3.6 Los Tipos de Grid

Grid ha establecido un nuevo paradigma para entregar información, recursos y servicios a los usuarios. Las implementaciones actuales

abarcan diversas aplicaciones en la industria y centros de investigación. En este creciente mundo de interconexiones se pueden distinguir tres categorías del Grid.

- Grid de Información
- Grid de Recursos
- Grid de Servicios

La figura 9 muestra la relación y la dependencia de estos tres Grids en cuanto al acceso uso y publicación de meta información.



Figura 9. Los tres tipos de Grid y sus relaciones conjuntas.

Fuente: <http://internetng.dit.upm.es/joe/Art/Gridfuturo.pdf>

### **3.6.1 Grid de Información**

Con la invención de la WWW en 1990, Tim Berners-Lee y Robert Calliau tomaron el primer y el más importante paso hacia la estructura global Grid. En solo unos años, el crecimiento exponencial de la Web creó una infraestructura de red pública y accesible, un Grid de Información omnipresente que entrega información de cualquier tipo a cualquier lugar en el mundo. Esta información se puede obtener al conectar cualquier computadora a una red telefónica pública vía modem, que es tan sencillo como enchufarse a la red eléctrica.

Servicios de compartición de archivos como Napster, Gnutella Network, E-Donkey forman parte del Grid de Información actual. A diferencia de la Web, los datos compartidos no se encuentran respaldados por una organización o dueño de algún sitio Web, sino que el servicio para compartir archivos es dispuesto por personas que desean intercambiar archivos de música, películas, videos o software. El servicio de intercambio se mantiene gracias a los participantes, no hay un repartidor central involucrado. Es un ambiente distribuido, dinámico y altamente flexible.

### **Situación Actual**

Desde su invención, se ha convertido en uno de los mayores éxitos en la tecnología y la comunicación, es usada por un gran número de la población mundial para acceder a información actual.

Una de las razones de su éxito es el concepto de hipervínculo, una referencia hacia otras webs que es muy fácil de usar. El seguir los hipervínculos es comúnmente la manera más rápida para encontrar información sin tener que teclear la información. Debido a esto, la Web dominó rápidamente a ftp y a redes que existían con anterioridad.

Otra causa para el éxito de esta red ha sido la facilidad con la que se puede actualizar la información, de esta forma ésta se mantiene al día.

### **3.6.2 Grid de Recursos**

Provee mecanismos para el uso coordinado de recursos como computadoras, archivos de datos, servicios e instrumentos de laboratorio. A diferencia del Grid de Información, usuarios anónimos no pueden acceder a este sin las credenciales necesarias a las facilidades y ventajas otorgadas por del Grid de Recursos. Solo usuarios autorizados y previamente registrados pueden utilizarlo.

La idea principal de este tipo de Grid es el proveer accesos sencillos, transparentes y eficientes a cualquier recurso independientemente de su localización. Estos recursos pueden ser desde poder de procesamiento, almacenamiento de información, ancho de banda, etc.

### **Situación Actual**

Son un poco más difíciles de implementar y desplegar, ya que los recursos son costosos y no pueden ser entregados al público en general sin costo alguno.

En esta categoría se encuentran los Grids Computacionales, que permiten el acceso a supercomputadores distribuidos para realizar tareas que consumen mucho tiempo. La mayoría se basan en la herramienta que se ha convertido en el estándar en esta área, el Globus Toolkit.

Hoy en día hay diversos prototipos de Grids específicos en investigación farmacéutica, química, astrofísica, tratamiento y representación de video, post producción, simulación del clima, geología, etc. Algunos de éstos utilizan portales Web, otros ocultan el acceso Grid dentro de la aplicación.

También están los Grids de Datos, los cuales proveen mecanismos para el almacenamiento seguro y redundante en sitios esparcidos geográficamente. Debido a los desafíos ocasionados al almacenar y procesar cantidades de Petabytes de datos en diferentes localizaciones, se convierte en un tema extendidamente demandante. Temas como la replicación, obtención, catalogación y la coordinación de estos datos deben ser aun refinados.

Otro ejemplo son las Aplicaciones paralelas como el ya mencionado proyecto SETI@home, fightcancer@home o distributed.net, los cuales han probado ser fácilmente ejecutables en computadores distribuidos. No necesitan Middleware ya que está integrado en la aplicación, la cual se encarga también de la ejecución de los trabajos remotos y la recolección de los resultados. Un aspecto importante de estas aplicaciones es la confianza implícita en ambas partes. El dueño del PC confía en la integridad del software sin verificar personalmente la autenticación y la autorización, y el

que distribuye el software Grid confía que los resultados no han sido falsificados por el dueño del PC. Los Grids de acceso también recaen dentro de esta categoría. Construyen la base técnica para la colaboración remota al proveer video conferencias interactivas y facilidades Blackboard.

### **3.6.3 Grid de Servicio**

Entrega servicios y aplicaciones sin importar la ubicación geográfica, implementación o plataforma de hardware. Los servicios son montados en los recursos concretos disponibles en el Grid de Recursos.

Una de las mayores diferencias entre estos dos tipos de Grid se encuentra en que el Grid de Servicios provee servicios abstractos sin importar su localización, mientras que el Grid de Recursos facilita accesos a recursos concretos ofrecidos en un sitio en particular.

### **Situación Actual**

Abarca los servicios disponibles como las maquinas de búsqueda, portales, paginas de servidor activas y diverso contenido dinámico.

Normalmente son gratuitos debido a patrocinios o publicidad. Servicios de email y autorización como Passport, GMX y Hotmail recaen en esta categoría.

Con los Web Services y el Open Grid Service Architecture OGSA están diseñados para proveer interoperabilidad entre los servicios sin importar la implementación, localización geográfica o plataforma de ejecución.

## **CAPITULO IV**

### **4. METODOLOGÍA DE LA INVESTIGACIÓN**

#### **4.1 Estado del Arte**

Existen varios casos exitosos de iniciativas de Grid a nivel mundial, se tomará como modelos dos de ellos:

Pragma (Pacific Rim Applications and Grid Middleware Assembly), es un grupo de instituciones dedicadas a generar aplicaciones que corran sobre un Grid. La ventaja y ganancia que tiene este grupo es que es el mismo donde nació una distribución de Linux preparada especialmente para uso en clusters y Grid Computing: Rocks. Estas instituciones han tenido inmensos avances en la puesta en marcha de clusters a nivel mundial, y a su vez, en la formación de Grids regionales.

SCRC (University Computing Centre at University of Zagreb), el centro de computo de la Universidad de Sarajevo es un ejemplo brillante de cómo una serie de instituciones se organizaron para construir un Grid nacional. Tienen instituidas diversas escuelas, tanto como de Linux, redes, programación MPI, entre otros, donde los profesionales son formados para trabajar en estos proyectos.

## **4.2 Propuesta de un Grid Universitario**

Los propósitos de un Grid Universitario son varios, como lograr un mayor poder de procesamiento para cálculos específicos, acopiar la mayor cantidad de información, disponibilidad de esta para los investigadores, etc. Es por eso que se busca un mayor acercamiento y cooperación entre las universidades para ganar y difundir conocimiento y experticia en esta tecnología, con el fin de incentivar y facilitar las investigaciones con requisitos importantes de cómputo.

## **4.3 Arquitectura a emplear: Open Grid Services Architecture (OGSA)**

OGSA define el esqueleto de lo que debe ser una arquitectura Grid, así como el modelo de programación de los servicios Grid. Suministra instrucciones de cómo construir un servicio Grid, con que componentes y la forma de ensamblarlos.

La GGF define OGSA [6] de esta manera: “Es un framework de amplio espectro de aplicación para la integración de Sistemas Distribuidos. Este framework define un núcleo de interfaces, comportamientos, modelos de recursos, enlaces, etc. que constituyen la plataforma OGSA.

### **4.3.2 Open Grid Services Infrastructure (OGSI)**

OGSI es la implementación de la infraestructura que define OGSA. Es lo que se llama el middleware o la plataforma para ejecutar servicios Grid. OGSI no determina que middleware concreto debe usarse y existen implementaciones OGSI en java, .Net, etc. Se puede utilizar tecnología Java 2



para crear y gestionar de servicios Grid e intercambiar información entre ellos de acuerdo a OGSi.

La definición que da el GGF de OGSi es esta: “OGSi define mecanismos para crear, gestionar e intercambiar información entre entidades llamadas Servicios Grid utilizando tecnología de Grid y de Web Services. Sucintamente, un Servicio Grid es un Web Service que se ajusta a una serie de convenciones (interfaces y comportamientos) que definen como los clientes interaccionan con el Servicio Grid. Estas convenciones y otros mecanismos OGSi asociados con la creación y localización (discovery) de servicios Grid permiten una gestión adecuada (control, fiabilidad, seguridad) de la información de estado distribuido y de larga duración que se requiere en la ejecución de aplicaciones distribuidas”.

#### **4.3.3 Web Services**

La base fundamental de los Servicios Grid en la implementación de Globus es el estándar Web Services. Web Services es una tecnología de computación de sistemas distribuidos que permite la creación de aplicaciones basándose en el modelo cliente/servidor mediante una plataforma independiente del lenguaje y utilizando protocolos abiertos como HTTP.

Los Web Services usan el protocolo SOAP (Simple Objects Access Protocol) y gramática XML para comunicarse, aunque el protocolo básico de transporte es HTTP para realizar las peticiones y las repuestas, aunque puede usar otros como JMS, etc. Mientras que las tecnologías distribuidas tales como CORBA, EJB, COM/DCOM, etc. están orientadas a clientes y servidores muy

dependientes, los Web Services están orientados usar clientes que no tienen un conocimiento previo del servicio hasta que se le invoca, es decir, existe un fuerte desacoplamiento entre cliente y servidor del servicio.

Los Servicios Web no manejan servicios con información de estado (stateful), es decir, no “recuerdan” valores de una llamada a otra. Sin embargo, en los entornos distribuidos contemplan en numerosas ocasiones escenarios en que si se requiere información de estado. Los Servicios Grid añaden esa capacidad a los Web Services.

#### **4.3.4 Grid Services**

Un Grid Service es una ampliación de los Web Services, su arquitectura está especificada por el Global Grid Forum (GGF). La idea era buscar una tecnología de objetos distribuidos que se adaptase a las necesidades de una aplicación Grid, y se pensó en los Web Services, aunque estos presentaban algunas limitaciones que se superaron:

- Los Web Services no mantienen el estado de una invocación a otra, los Grid Services sí.
- Los Web Services no son transientes, es decir no se pueden crear varias instancias de un mismo servicio según se necesita y destruirlas cuando ya no son necesarias, en los Grid Services, se puede.
- Los Web Services no incluyen servicios de apoyo que han sido incluidos en los Grid Services como son las notificaciones, el servicio de persistencia, la gestión del ciclo de vida, etc.

Los Grid Services utilizan un enfoque de Factorías de Objetos de manera que en lugar de tener un único servicio compartido por todos los usuarios (como el Web Service) se tiene un servicio factoría que crea instancias individuales del servicio. Cuando se invoca a una operación del servicio se accede a la instancia y no a la factoría.

Además se puede crear una instancia por cliente, o varias por cliente o una para varios clientes. Por último la destrucción de la instancia puede correr a cargo del cliente o de la factoría. Los Grid Services están contruidos utilizando Web Services pero se les incorporan una serie de mecanismos de la plataforma OGSi. Estos mecanismos adicionales se pueden agrupar en cuatro áreas:

- Servicio de Nombres (Naming), que asegura la existencia de un nombre único para cada instancia de Servicio Grid y permite la localización de Grid Services (discovering) mediante nombres.
- Servicio de Datos, que gestiona los conjuntos de datos asociados a la ejecución de un servicio Grid.
- Notificación, es decir el conjunto de interfaces para registrar y suministrar notificaciones y subscripciones. Estos son los mecanismos usados para la comunicación entre los componentes de una aplicación Grid.
- Ciclo de Vida, mecanismos para la creación y destrucción de instancias de Servicios Grid.

#### **4.3.4 Arquitectura de OGSA**

Cuatro capas principales comprenden la arquitectura de OGSA, empezando desde la inferior, ellas son:

- Recursos: los recursos físicos y los recursos lógicos Web Services más las extensiones de OGSF que definen servicios Grid.

- Servicios de arquitectura de OGSA.

- Aplicaciones Grid.

Se observaran estas capas, una a la vez.

##### **4.3.1.1 Capa de los recursos físicos y lógicos**

El concepto de recursos es central para OGSA y Grid Computing en general. Los recursos comprenden las capacidades del Grid, y no son limitados a los procesadores. Los recursos físicos incluyen servidores, almacenamiento, y red.

Sobre los recursos físicos están los recursos lógicos. Ellos proporcionan función adicional virtualizando y agregando los recursos en la capa física. El propósito general del software intermedio tal como sistemas de archivos, gestores de bases de datos directorios, y gestores de flujos de trabajo (workflow) es proporcionar estos servicios abstractos sobre el Grid físico.

##### **4.3.1.2 Capa de Web Services**

La segunda capa en la arquitectura de OGSA es la de Web Services.

Aquí hay un importante principio de OGSA: Todos los recursos del Grid (lógico y físico) son modelados como servicios. La especificación Abierta de Infraestructura de Servicios de Grid (OGSI) define servicios de Grid y construye sobre las tecnologías de Web Services Standard. OGSI aprovecha los mecanismos de Web Services como XML y WSDL para especificar interfaces estándares, conductas e interacción para todos los recursos del Grid.

OGSI extiende la definición de Web Services para proporcionar capacidades para una Web Service dinámica, estable y manejable que se exige para modelar los recursos del Grid.

#### **4.3.1.3 Capa Servicios de Grid de la Arquitectura de OGSA**

La capa de Web Services, con sus extensiones de OGSI, provee una infraestructura base a la próxima capa de arquitectura de Grid Services. El Global Grid Forum está actualmente trabajando para definir muchos de estos servicios de Grid de arquitectura en áreas como la ejecución de programas, servicios de datos, y servicios centrales. Algunos ya se han definido, y algunas aplicaciones ya han aparecido. En tanto las aplicaciones de éstos servicios nuevos de arquitectura empiecen a aparecer, OGSA se volverá una arquitectura orientada al servicio más útil (SOA).

#### **4.3.1.4 Capa de Aplicaciones de Grid**

Con el tiempo, en tanto una cantidad importante de servicios de arquitectura de Grid continúe desarrollándose, aparecerán las nuevas aplicaciones del Grid que utilicen uno o más servicios de arquitectura de éste. Estas aplicaciones comprenden la cuarta capa principal de la arquitectura de OGSA.

#### **4.4 Herramienta a emplear: Globus Toolkit**

Globus Toolkit [6] es un conjunto de herramientas de software de código fuente abierto, utilizado para construir Grids. Esta siendo desarrollado por Globus Alliance y muchos otros por todo el mundo. Un número creciente de proyectos y compañías que están utilizando Globus Toolkit para aprovechar el potencial del Grid Computing.

La Globus Alliance está desarrollando fundamentalmente tecnologías necesarias para construir computación Grid.

Globus Toolkit incluye servicios de software y librerías para el monitoreo, descubrimiento y administración de recursos, más seguridad y administración de archivos. En su última versión, GT3, es la primera implementación a escala completa de las especificaciones de Arquitectura abierta de servicios Grid (Open Grid Service Arquitectura (OGSA)) en la cual la Globus Alliance está jugando un rol importante en su definición.

Los Pilares básicos del Globos Toolkit, desde la versión GT2 son los que se mencionan a continuación e ilustran en la Figura 10.

##### **4.4.1 Seguridad**

Globus Toolkit utiliza la infraestructura de seguridad Grid (Grid Security Infrastructure, GSI) para permitir autenticación u comunicación segura sobre una red de trabajo. GSI provee un número útil de servicios para Grids, incluyendo autenticación a sola firma.

GSI está basado sobre encriptación de clave pública, certificados X.509, y servicios de protocolos de comunicación SSL (Secure sockets Layer), Extensiones a estos estándares han sido agregadas para incluir autenticación de tipo a sola firma y delegación. La implementación de Globus Toolkit de GSI incorpora API de servicio de seguridad genérico (GSS-API), la cual es un API estándar para los sistemas de seguridad promovidos por la fuerza de tarea en ingeniería de Internet (IETF).

#### **4.4.2 Componentes de administración de datos**

GridFTP es protocolo de alto rendimiento, seguridad, confiabilidad de transferencia de datos optimizado para importantes anchos de banda en redes de extensión amplia. El GridFTP está basado sobre FTP, el más popular protocolo de transferencia de archivos de Internet.

#### **4.4.3 Administración de recursos**

El servicio administrador y localizador de recursos Grid (Globos Resource Allocation Manager, GRAM) provee una simple interfaz para el requerimiento y uso de sistemas de recursos remotos para la ejecución de trabajos. Está diseñado para proveer una flexible interfaz a sistemas de planificación de trabajos.

Globus Toolkit 3.2 contiene dos implementaciones GRAM, una basada en un propietario, el protocolo de servicios pre-Web (Pre-WS GRAM) y un segundo construido utilizando interfaz de servicios Web (WS GRAM).

#### 4.4.4 Servicios de información

El sistema de monitoreo y descubrimiento (Meta-computing Directory Service, MDS) es el componente de servicio de información de Globus Toolkit y provee información sobre recursos disponibles sobre la Grid y su estado.

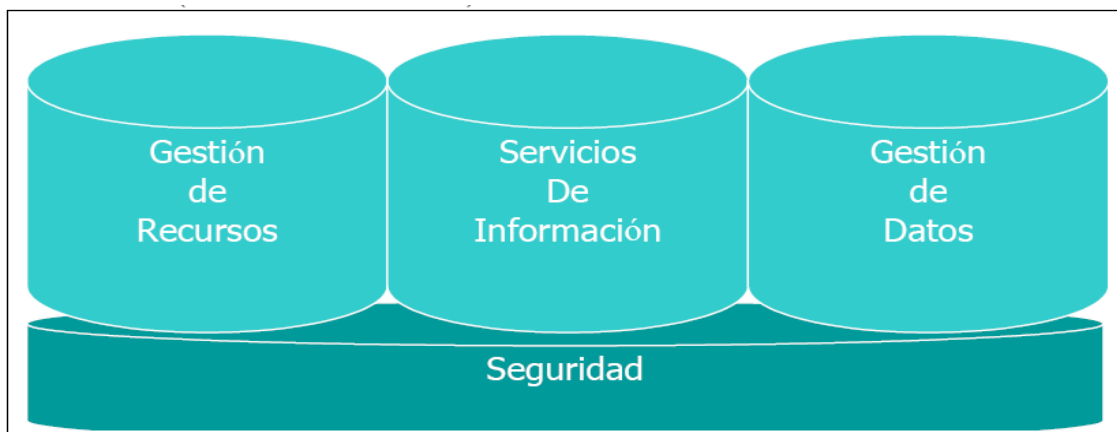


Figura 10. Componentes Fundamentales del Globus Toolkit.

Fuente: <http://internetng.dit.upm.es/joe/Art/Globus.pdf>



#### **4.5 Arquitectura Conceptual del Grid Universitario**

El diseño de la arquitectura de un Grid universitario que sea viable, incluye mucho más que agregar el poder de procesamiento de las máquinas ociosas vía software de las diferentes universidades, hay que tener en cuenta que un ambiente Grid debe ser capaz de interoperar con una amplia variedad de tecnología que están emergiendo.

Se propone una arquitectura conceptual del Grid universitario, basándonos en el estándar OGSA, para esto tenemos en cuenta los siguientes pasos [11]:

1. Integrar componentes de SW y HW en un recurso de red combinados.
2. La implementación de un Middleware para proveer una visión transparente de los recursos disponibles.
3. El desarrollo de herramientas que permitan la administración y control de la infraestructura y las aplicaciones.
4. El desarrollo y optimización de aplicaciones distribuidas que permiten hacer un uso optimo de infraestructura y recursos disponibles.

Es así que proponemos una arquitectura del Grid Universitario tal y como se muestra en la Figura 11.

En la capa de infraestructura, tendremos en cuenta que nuestros entornos universitarios están repletos de recursos computacionales subutilizados, esto los beneficia directamente ya que no incurrirán en gastos de

infraestructura, además se cuenta con una Red Académica Avanzada Peruana (RAAP), la cual está formada por varias universidades locales.

En la capa de servicios, el Globus Toolkit, que incluye servicios de software y librerías para el monitoreo, descubrimiento y administración de recursos, más seguridad y administración de archivos. En su última versión, GT3, es la primera implementación a escala completa de las especificaciones de Arquitectura abierta de servicios Grid (Open Grid Service Arquitectura (OGSA)).

Finalmente en la capa de aplicación, estas se irán implementando de acuerdo a las necesidades o problemáticas que puedan tomar ventaja del Grid.

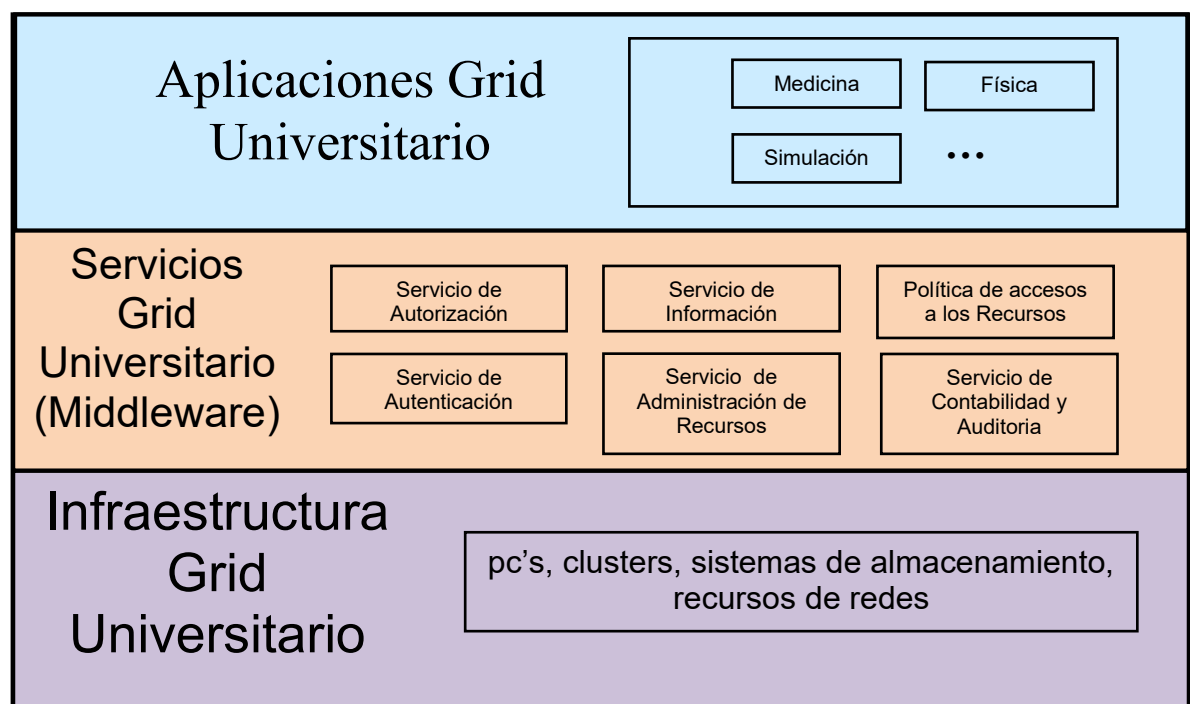


Figura 11. Arquitectura Conceptual del Grid Universitario

## **4.6 Implementación del Grid Universitario**

### **4.6.1 Integrantes del Grid universitario:**

Para llevar a cabo esta propuesta de creación del primer Grid Universitario Peruano, consideraremos como los primeros integrantes de este Grid a las universidades Nacional de Ingeniería (UNI), Nacional Mayor de San Marcos (UNMSM) y Nacional Agraria la Molina (UNALM), dado que existe una Alianza Estratégica entre estas universidades y por ende el compromiso de desarrollar programas de mejoramiento del nivel de las investigaciones.

### **4.6.2 Estructura del Grid Universitario:**

#### **4.6.2.1 Hardware / Recursos:**

Nuestros entornos están repletos de recursos computacionales subutilizados, se sabe que en los ambientes universitarios, existen múltiples laboratorios de computadoras, además de los equipos que están asignados a docentes, secretarías, etc. Teniendo en cuenta que la fracción de tiempo que estos equipos son usados efectivamente es sumamente pequeño.

Por todo esto no es necesario, por parte de las universidades participantes, una inversión en hardware. Teniendo en cuenta que el hardware de un Grid puede ser cualquier dispositivo o equipo con capacidades de comunicación a través de una red.

#### **4.6.2.2 Software:**

Para la implementación del Grid universitario se usará el Globus Toolkit, dado que se ha convertido en el estándar “de facto” y es de licencia libre.

Recordemos que el software de un Grid está compuesto por un middleware, que sirve de enlace entre los usuarios o aplicaciones y los recursos. Tal y como se muestra en la Figura 12. Este software es la clave para la implementación del Grid.

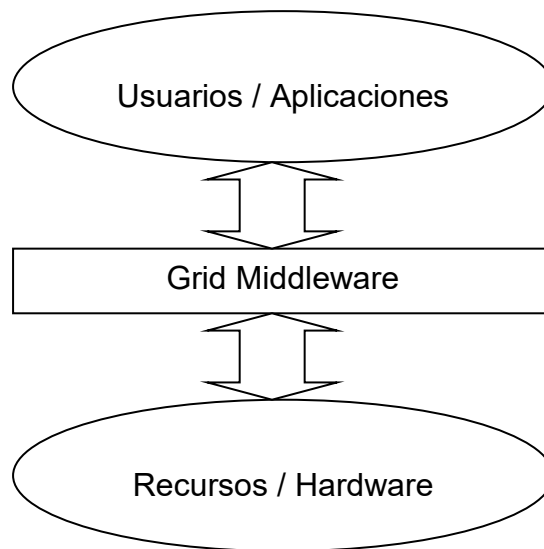


Figura 12. Ubicación del Middleware

#### **4.6.3.3 Usuarios / Aplicaciones:**

Las universidades participantes en este caso la UNI, la UNMSM y la UNALM y sus respectivas áreas de investigación irán desarrollando aplicaciones de acuerdo a la problemática y sus necesidades.

#### **4.6.3 Etapas del Grid Universitario**

El Grid universitario se desarrollara en varias etapas. Inicialmente estará integrado por la UNI, la UNMSM y la UNALM y en la medida que se resuelvan inconvenientes y se domine el problema se integraran otras instituciones.

#### 4.6.3.1 Primera Etapa:

Es muy importante que las instituciones participantes se organicen en equipos, se debería contar con un equipo de Análisis y un equipo Técnico de Soporte y además se recomienda la contratación de un consultor experto en implementaciones de Grid Computing, que oriente y les transfiera los conocimientos necesarios a estos equipos.

La primera etapa, consistiría en lograr la integración de la Universidad Nacional de Ingeniería y la de la Universidad Nacional Mayor de San Marcos, con una estructura similar a la que se muestra en la Figura 13.

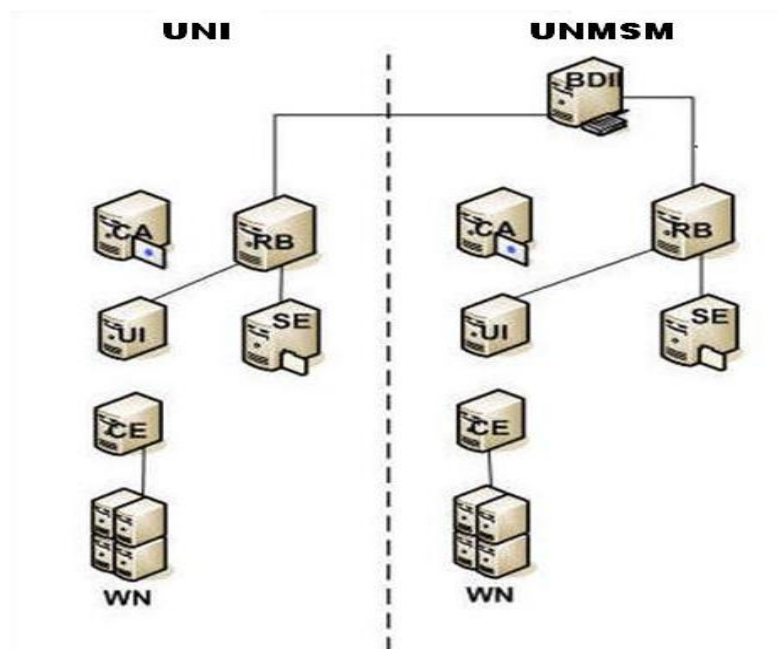


Figura 13. Grid UNI – UNMSM

#### 4.6.3.2 Segunda Etapa:

La segunda etapa, se iniciaría con la integración de la Universidad Nacional Agraria la Molina con el Grid UNI-UNMSM, quedando con una estructura similar a la que se muestra en la Figura 14. Resaltaremos que para usar un Grid no es necesario tener un sitio. La entrada a un Grid es a través de una Interfaz de Usuario (UI). Una institución que en un momento dado no cuenta con recursos o no disponga de la capacidad técnica para montar un sitio, pueden acceder al Grid a través del UI de otra institución.

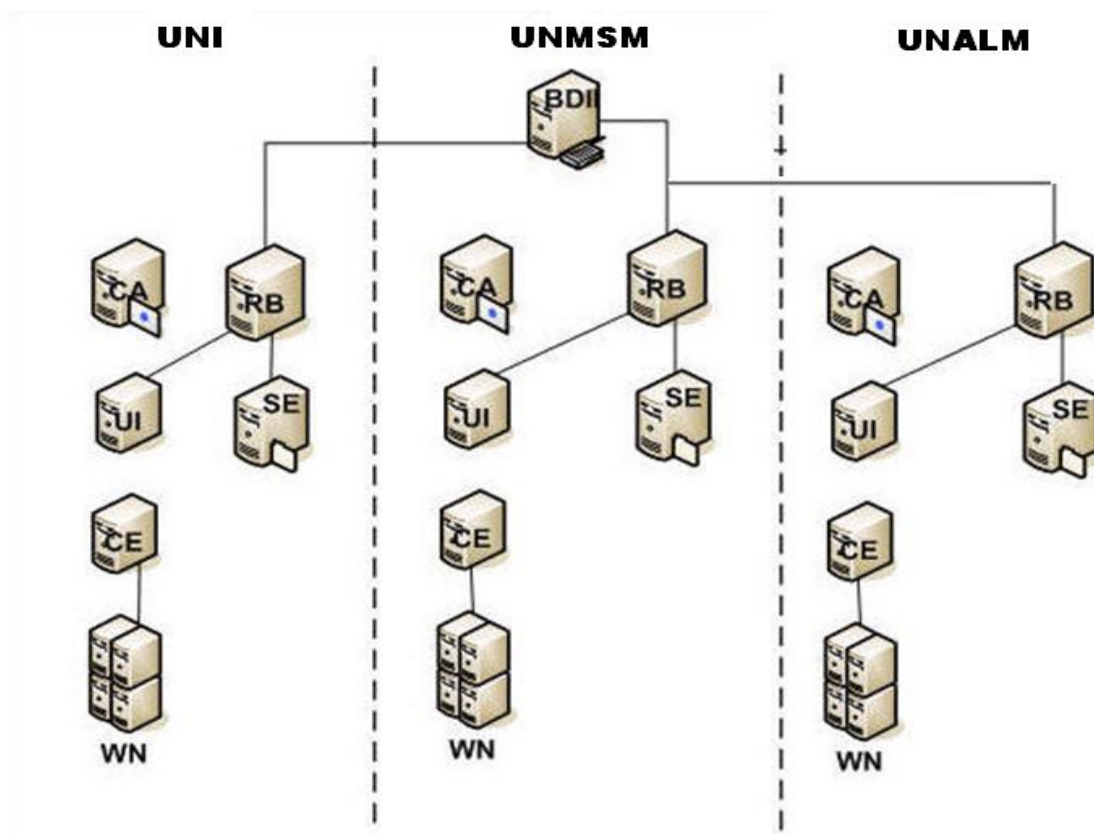


Figura 14. Grid UNI-UNMSM-UNALM

#### 4.6.3.3 Tercera Etapa:

La tercera etapa, sería implementar soluciones a problemáticas definidas que puedan tomar ventaja del Grid, tales como: los problemas de bioinformática, simulaciones climáticas, o cualquier otro problema científico que requiera de muchos recursos y que pueda ser distribuido.

#### 4.6.3.4 Cuarta Etapa:

Con todo esto hecho, la cuarta etapa, empezaría por elaborar un proyecto universitario a nivel de Lima – Metropolitana (Ver Figura 15.), que financie la continuidad de esta iniciativa, mostrando los éxitos ya obtenidos a través del Grid UNI-UNMSM-UNALM y, con eso, dar un impulso final al objetivo principal: promover el uso de Grid Computing como herramienta útil para solucionar problemas de gran escala. En la figura 16 se muestra el Grid Universitario sobre RAAP,

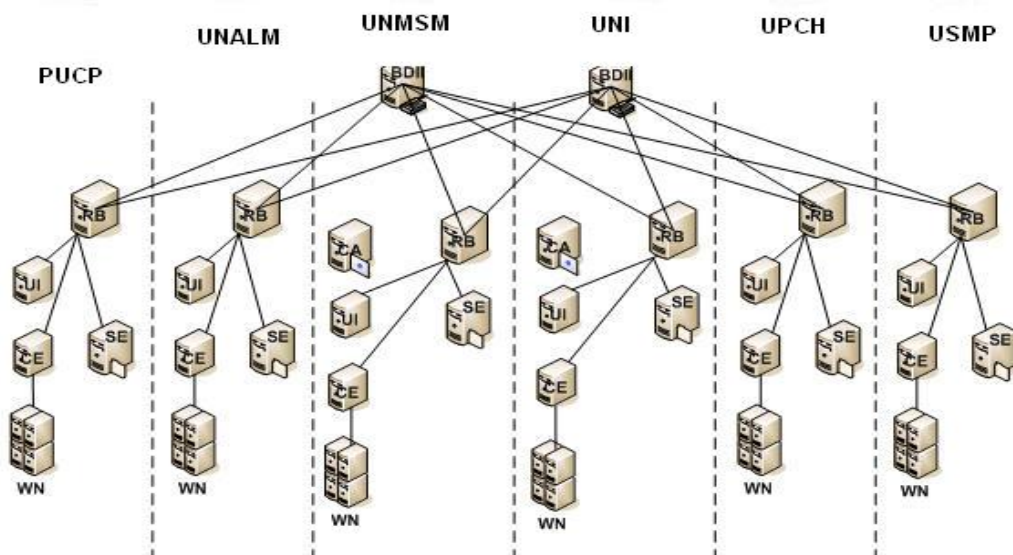


Figura 15. Grid Lima-Metropolitana

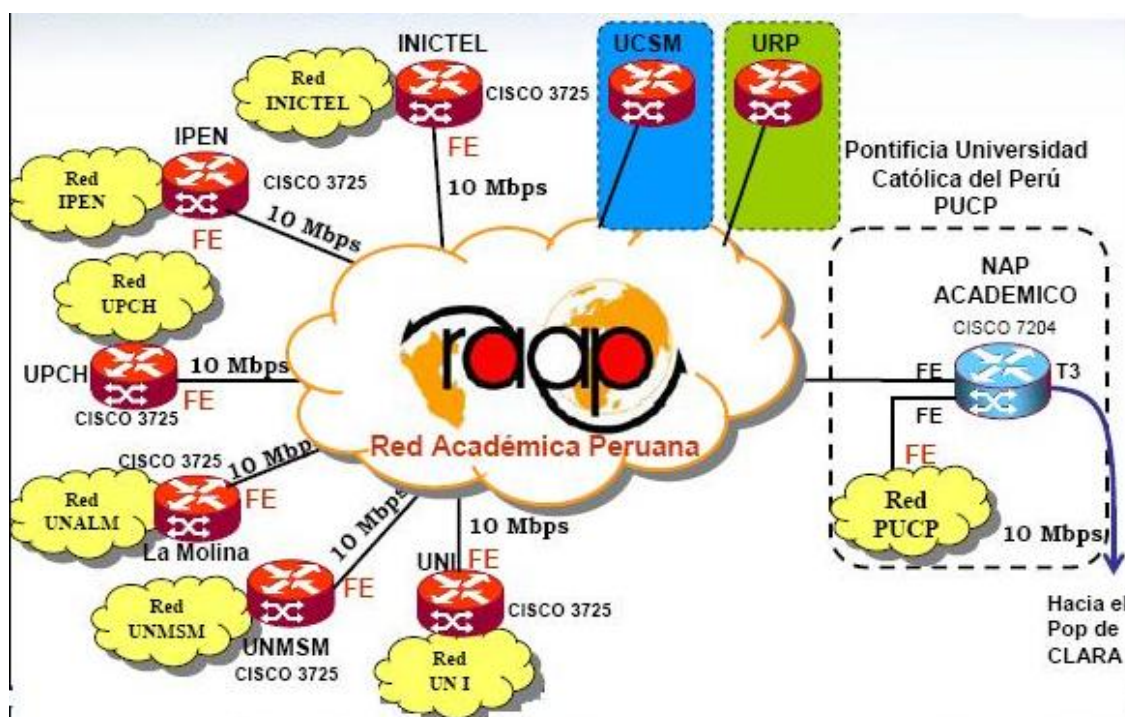


Figura 16. Grid Lima-Metropolitana sobre RAAP

Fuente: <http://www.raap.org>

#### 4.7 Evaluación y Análisis

Para participar en esta propuesta se necesita fundamentalmente compromiso y dedicación. La idea es que las instituciones participantes destinen recursos (humanos y tecnológicos) para trabajar en la implementación de este Grid universitario.

Lo importante de esta iniciativa es la generación de conocimiento que después permita ofrecer a nuestros investigadores una plataforma científica atractiva, homogénea, descentralizada y a la medida de cada necesidad.



## **CONCLUSIONES**

1. El Grid universitario beneficiaría a todos los centros de investigación universitarios, y permitirá lograr avances a pasos agigantados en sus proyectos de investigación científica.
2. El Grid universitario ayudara a las universidades participantes para que no incurran en gastos adicionales de infraestructura en el desarrollo de sus proyectos científicos.
3. Con el Grid se incrementara la capacidad de recursos en todas las instituciones participantes.

## **RECOMENDACIONES**

1. Para una rápida implementación y desarrollo de un Grid Nacional sería recomendable extender la invitación a universidades e instituciones del interior del país.
2. La Universidad Nacional Mayor de San Marcos con el apoyo de la Facultad de Ingeniería de Sistemas, debe ser la pionera en fomentar el uso del Grid Computing para el desarrollo de proyectos científicos a nivel nacional.
3. Los investigadores independientes, ya sean personas u otras instituciones, deberían de tener acceso a la infraestructura del Grid universitario, una vez implementado para así apoyarlos en sus investigaciones

## **REFERENCIAS BIBLIOGRÁFICAS**

[1] Berstis, Viktors

2002            Fundamentals of Grid Computing. IBM, USA 2002

[2] Foster I., Kesselman C. y Tuecke S.

2001            The Anatomy of the Grid: Enabling Scalable  
Virtual Organization. USA 2002.

[3] Sotomayor, B.

2004            Introducción a la Computación Grid,  
30 de abril del 2004.

[4] Losilla Anadón, G.

2005            Introducción al Grid. España 2005

[5] Doallo R., Touriño J y Heymann E.

2006            Tecnología Grid,  
25 de agosto del 2006

[6] The Globus Alliance

2007      <http://www.globus.org/>

[7] Barrios, Verónica Vanessa

2006      Grid Computing. Trabajo Final de Aplicación. Universidad  
Nacional del Nordeste, Argentina.

[8] Grid Café

2007      <http://gridcafe.web.cern.ch/gridcafe/>

[9] J. Nick S. Tuecke I. Foster, C. Kesselman.

2002      The Physiology of the Grid: An Open Grid Services Architecture  
for Distributed Systems Integration. Global Grid Forum, USA,  
2002.

[10] T.Durniak, P.Herman, J.Karuturi, C.Woods, C.Gilman, J.Barry y M.Aparicio.

1998      Enterprise Distributed Computing Workshop. IEEE, USA, 1998.

[11] Erwin Mac Dowall Reynoso.

Middleware. Universidad Federal Fluminense, Brasil.

## **TERMINOLOGIA**

### **Autenticidad**

La autenticación de un objeto significa la confirmación de su procedencia, mientras que la autenticación de una persona a menudo consiste en verificar su identidad.

### **Autoridad de Certificación**

Una Autoridad de Certificación (AC o CA por sus siglas en inglés Certification Authority) es una entidad de confianza, responsable de emitir y revocar los certificados digitales, utilizados en la firma digital, para lo cual se emplea la criptografía de clave pública.

### **Certificado Digital**

Un certificado digital es un documento digital mediante el cual un tercero confiable (una autoridad de certificación) garantiza la vinculación entre la identidad de un sujeto o entidad y su clave pública.

## **Clusters**

Un cluster de computadoras es un grupo de computadoras acopladas que trabajan juntas y proveyendo una visión de computadora única. Comúnmente las computadoras del cluster están conectadas a través de una red local. Los clusters mejoran la performance y la disponibilidad de recursos, en comparación a la prestación de una sola computadora.

## **Confidencialidad**

Confidencialidad es la propiedad de un documento o mensaje que únicamente está autorizado para ser leído o entendido por algunas personas o entidades. Se dice que un documento o mensaje es confidencial si éste sólo está autorizado a ser leído o entendido por un destinatario designado.

## **FLOPS**

En informática, acrónimo de Floating point Operations Per Second (operaciones de punto flotante por segundo).

## **Globus Toolkit**

Es un conjunto de herramientas de código fuente abierto, desarrolladas por Globus Alliance, usado para construir infraestructura y aplicaciones Grid.

## **Grid Computing**

Es un tipo de sistema distribuido basado en computadoras conectadas por una red (usualmente Internet) para resolver problemas de cómputo

intensivo. Es opuesto a la noción de supercomputadora, que posee múltiples CPUs conectadas a través de un bus local de alta velocidad.

### **GridFTP**

Es un protocolo de transferencia de datos, de alta performance, seguro y confiable para redes de área amplia de alta velocidad, basado en el protocolo de Internet FTP (File Transfer Protocol). Provee extensiones para operaciones de alta performance. Está desarrollado por Globus Alliance.

### **Hash**

Una función de hash es una función para resumir o identificar probabilísticamente un gran conjunto de información, dando como resultado un conjunto imagen finito generalmente menor. En el caso de la información digital, el hash resultante es una cadena de bits de longitud fija.

### **Integridad**

La integridad de datos significa garantizar que los mismos no han sido alterados y conservan su estado original.

### **Middleware**

Es un software que conecta aplicaciones o componentes de software (básicamente a nivel de sistema operativo). Es usado para proveer soporte a aplicaciones distribuidas, entre ellas servidores web, servidores de aplicaciones, sistemas de gerenciamiento de contenido, que son usadas para desarrollo y deployment de aplicaciones. Los middlewares están integrados a

tecnologías actuales, basadas en XML (eXtensible Markup Language), SOAP (Simple Object Access Protocol), Web Services y arquitecturas basadas en servicios.

### **MPI (Message Passing Interface)**

Es una librería de comunicaciones independiente del lenguaje, que puede usarse para programar sobre computadoras paralelas.

### **OGSA (Open Grid Service Architecture)**

Describe una arquitectura para entornos de Grid Computing orientado a servicios, para uso científico y empresarial. Está basado en tecnologías de Web Service como WSDL (Web Services Description Language) y SOAP (Simple Object Access Protocol) y, desarrollado por Global Grid Forum.

### **Round - Robin**

Es un método para seleccionar todos los elementos en un grupo de manera equitativa y en un orden racional, normalmente comenzando por el primer elemento de la lista hasta llegar al último y empezando de nuevo desde el primer elemento. El planeamiento Round Robin es tan simple como fácil de implementar, y está libre de inanición.

### **Schedulers**

El planificador es un componente funcional muy importante de los sistemas operativos multitarea y multiproceso, y es esencial en los sistemas operativos de tiempo real. Su función consiste en repartir el tiempo disponible



de un microprocesador entre todos los procesos que están disponibles para su ejecución.

### **Sistema Paralelo**

Es la combinación de un algoritmo paralelo y la máquina paralela sobre la que éste se ejecuta. Una máquina paralela tiene dos o más procesadores, y un algoritmo paralelo es la especificación de los pasos a realizar por cada procesador más las acciones de comunicación y sincronización. Existen diferentes tipos de sistemas paralelos, dependiendo de las distintas combinaciones entre ambos elementos.

### **Sistemas Distribuidos**

Conocidos como computación distribuida, es un modelo usado para resolver problemas de computación masiva utilizando un gran número de computadoras organizadas en clusters incrustados en una infraestructura de comunicaciones distribuida. La computación en clusters consiste en compartir recursos heterogéneos (basadas en distintas plataformas, arquitecturas de equipos y programas, lenguajes de programación), situados en distintos lugares y pertenecientes a diferentes dominios de administración sobre una red que utiliza estándares abiertos. Sintéticamente, consiste en virtualizar los recursos informáticos.

## **Web Services**

Es un módulo de software que se hace disponible a través de Internet usando protocolos y estándares abiertos, basados en XML.

## **Virtualización**

Es una tecnología que permite progresar en la creación de una capa de abstracción que independiza el nivel funcional o de aplicación del hardware existente.